Package 'PALM'

January 20, 2025

Type Package

Title Fast and reliable association discovery in large-scale microbiome studies and meta-analyses

Version 0.1.0

Author Zhoujingpeng Wei

Maintainer Zhoujingpeng Wei <zwei74@wisc.edu>

Description This R package implements a quasi-Poisson-based framework designed for robust, scalable, and reproducible identification of covariate-associated microbial features in large-scale microbiome association studies and meta-analyses.

Data 2024-11-23

Reference Wei et al. 'Fast and reliable association discovery in large-scale microbiome studies and meta-analyses using PALM'. Submitted.

License GPL (>= 3) | file LICENSE

LinkingTo Rcpp, RcppArmadillo

Imports MASS, dplyr, brglm2, Rcpp

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

Suggests knitr, rmarkdown

Depends R (>= 4.3.0)

VignetteBuilder knitr

R topics documented:

CRC_data *Datasets from two metagenomics studies of colorectal cancer (CRC)*

Description

The "CRC_data" includes "CRC_abd" and "CRC_meta". "CRC_abd" is a sample-by-feature matrix of relative abundance counts from the two studies including 267 species under order "Clostridiales". The "CRC_meta" is a data frame including the sample-level variables from the two studies.

Usage

```
data(CRC_data)
```

Format

An object of class `list` of length 2.

Source

<https://github.com/zellerlab/crc_meta>

References

Wirbel, Jakob et al. Nat Med. 2019 Apr;25(4):679-689.

Examples

```
library("PALM")
data("CRC_data")
CRC_abd <- CRC_data$CRC_abd
CRC_meta <- CRC_data$CRC_meta
```

palm *Meta-analysis for feature-level association testing for each covariate of interest.*

Description

Implementation of PALM for single-study association analysis and meta-analysis across multiple studies. This function conducts feature-level association testing, evaluating one covariate of interest at a time.

Usage

```
palm(
  rel.abd,
  covariate.interest,
  covariate.adjust = NULL,
  cluster = NULL,
  depth = NULL,
  depth.filter = 0,
  prev.filter = 0.1,
  p.adjust.method = "fdr",
  correct = TRUE
)
```

Arguments

rel.abd	For a single association study, provide a matrix of relative abundance counts, with sample IDs as row names and microbial feature IDs as column names. The matrix must not contain any missing values. For a meta-analysis, provide a list of such matrices, where each element corresponds to one study, and the name of each element represents the study ID.
covariate.interest	For a single association study, provide a matrix where rows represent samples, and columns represent numeric covariates of interest. The order of samples must match the order in "rel.abd", and the matrix must not contain any missing values. For a meta-analysis, provide a list of such matrices, where each element corresponds to one study, and the name of each element represents the study ID. The set of covariates can differ across studies.
covariate.adjust	For a single association study, provide a data frame where rows represent samples, and columns represent covariates to be adjusted in the model. Covariates must be either factors or numeric vectors. The order of samples must match the order in rel.abd, and the data frame must not contain any missing values. For a meta-analysis, provide a list of such data frames, where each element corresponds to one study, and the name of each element represents the study ID. The set of covariates can differ across studies. The default is NULL, meaning no covariates are adjusted.
cluster	For a single association study with correlated samples, provide a vector that defines the sample clusters. The order of samples in the vector must match the order in "rel.abd". For example, the values of this variable can be subject IDs if each subject has multiple correlated samples (e.g., in a longitudinal study). For a meta-analysis, provide a list of such vectors, where each element pertains to one study with correlated samples, and the name of each element must be the study ID. The order of samples in each vector must match the order in "rel.abd" for the corresponding study. The default is NULL, assuming all samples across all studies are independent, in which case this input is not required.
depth	For a single association study, provide a vector representing the sequencing depth for each sample. The length of the vector must match the number of samples, and the order must align with "rel.abd". For a meta-analysis, provide a list of such vectors, where each element corresponds to one study, and the name of each element must be the study ID. The length of each vector must match the sample size of the corresponding study. The default is NULL, in which case the sequencing depth is calculated as the row sum of "rel.abd".

`depth.filter` A cutoff value to remove samples with sequencing depth less than or equal to the cutoff. Default is 0.

`prev.filter` A cutoff value remove microbial features with prevalence (proportion of nonzero observations) less than or equal to the cutoff. This cutoff is applied to each study. So, a feature could be removed in a subset of the studies. The cutoff value must be in the range of 0-1. Default is 0.1.

`p.adjust.method` p-value correction method, a character string, should be one of "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none". Default is "fdr".

`correct` This argument takes the default value of TRUE. If TRUE, it will correct the compositional effect; If FALSE, it will not correct the compositional effect (i.e., directly use RA summary statistic to do analysis and output RA-level summary statistics).

Value

Output a list with each component for a covariate of interest.

For the single-study analysis, the component includes the following elements:

`palm_single` A data framework containing association effect estimates, standard errors, p-values, and q-values and summary statistics (association effect estimates and standard errors).

For the met-analysis of multiple association studies, the component includes the following elements:

`palm_meta` A data framework containing overall association effect estimates, standard errors, p-values, and q-values for testing the overall effect; p-values and q-values for testing cross-study effect heterogeneity; and summary statistics (association effect estimates and standard errors) for individual studies.

Author(s)

Zhoujingpeng Wei <zwei74@wisc.edu>.

See Also

`palm.get.summary`, `palm.null.model`, `palm.meta.summary`

Examples

```
library("PALM")
data("CRC_data", package = "PALM")
CRC_abd <- CRC_data$CRC_abd
CRC_meta <- CRC_data$CRC_meta

########## Generate summary statistics ##########
rel.abd <- list()
covariate.interest <- list()
for(d in unique(CRC_meta$Study)){
  rel.abd[[d]] <- CRC_abd[CRC_meta$Sample_ID[CRC_meta$Study == d],]
  disease <- as.numeric(CRC_meta$Group[CRC_meta$Study == d] == "CRC")
  names(disease) <- CRC_meta$Sample_ID[CRC_meta$Study == d]
 covariate.interest[[d]] <- matrix(disease, ncol = 1, dimnames = list(names(disease), "disease"))
}
```

```
meta.result <- palm(rel.abd = rel.abd, covariate.interest = covariate.interest)
```

`palm.get.summary` *Generate palm summary statistics for individual studies.*

Description

This function directly takes the output object from the "palm.null.model" function and constructs summary statistics for covariates of interest in each study. The output can be directly used by the function "plam.meta.summary" to perform meta-analysis.

Usage

```
palm.get.summary(null.obj, covariate.interest, cluster = NULL, correct = TRUE)
```

Arguments

`null.obj`	The output of function "palm.null.model".
`...`	See function palm. In "covariate.interest" and "cluster", the order of samples should be matched with the order in "rel.abd" used in the "palm.null.model" function.

Value

Output a list with each component for a study. The component includes the following elements.

`est`	A matrix contains the association effect estimates for the study. The row names are microbial feature IDs and the column names are the covariates of interest IDs.
`stderr`	A matrix contains the standard errors of the association effect estimates for the study. The row names are microbial feature IDs and the column names are the covariates of interest IDs.
`n`	Sample size for the study.

See Also

`palm.null.model`, `palm.meta.summary`, `palm`

Examples

```
library("PALM")
data("CRC_data", package = "PALM")
CRC_abd <- CRC_data$CRC_abd
CRC_meta <- CRC_data$CRC_meta

########## Generate summary statistics ##########
rel.abd <- list()
covariate.interest <- list()
for(d in unique(CRC_meta$Study)){
```

```
  rel.abd[[d]] <- CRC_abd[CRC_meta$Sample_ID[CRC_meta$Study == d],]
  disease <- as.numeric(CRC_meta$Group[CRC_meta$Study == d] == "CRC")
  names(disease) <- CRC_meta$Sample_ID[CRC_meta$Study == d]
 covariate.interest[[d]] <- matrix(disease, ncol = 1, dimnames = list(names(disease), "disease"))
}

 null.obj <- palm.null.model(rel.abd = rel.abd)

 summary.stats <- palm.get.summary(null.obj = null.obj, covariate.interest = covariate.interest)
```

`palm.meta.summary` *Meta-analyze summary statistics across studies*

Description

This function directly takes the summary statistics output from the "palm.get.summary" function and combines the summary statistics across studies for feature-level association testing for each covariate of interest.

Usage

```
palm.meta.summary(summary.stats = summary.stats, p.adjust.method = "fdr")
```

Arguments

`summary.stats`	The output of function "palm.get.summary".
`...`	See function palm.

Value

Same output as the function "palm" when performing meta-analysis.

See Also

`palm.null.model`, `palm.get.summary`, `palm`

Examples

```
library("PALM")
data("CRC_data", package = "PALM")
CRC_abd <- CRC_data$CRC_abd
CRC_meta <- CRC_data$CRC_meta

########## Generate summary statistics ##########
rel.abd <- list()
covariate.interest <- list()
for(d in unique(CRC_meta$Study)){
  rel.abd[[d]] <- CRC_abd[CRC_meta$Sample_ID[CRC_meta$Study == d],]
  disease <- as.numeric(CRC_meta$Group[CRC_meta$Study == d] == "CRC")
  names(disease) <- CRC_meta$Sample_ID[CRC_meta$Study == d]
 covariate.interest[[d]] <- matrix(disease, ncol = 1, dimnames = list(names(disease), "disease"))
```

```
}

null.obj <- palm.null.model(rel.abd = rel.abd)

summary.stats <- palm.get.summary(null.obj = null.obj, covariate.interest = covariate.interest)

########## Meta-analysis ##########
meta.result <- palm.meta.summary(summary.stats = summary.stats)
```

`palm.null.model` *Get information from the null model.*

Description

The "palm.null.model" function computes the estimated mean microbial feature proportions and residuals of the relative abundance under the null model of no associations for each study (so this function does not need the covariate.interest information). The output of this function will be fed into the "palm.get.summary" function to construct summary statistics for covariates of interest in each study.

Usage

```
palm.null.model(
  rel.abd,
  covariate.adjust = NULL,
  depth = NULL,
  depth.filter = 0,
  prev.filter = 0.1
)
```

Arguments

`...`	See function palm

Value

Output a list with each component for a study. The component includes the following elements.

`Y_I`	A matrix of predicted abundance.
`Y_R`	A matrix of Firth's bias corrected abundance residual.
`Z`	Cleaned ad formatted covariate.adjust.
`rm.sample.idx`	The index of the removed samples for the study.

See Also

`palm.get.summary`, `palm.meta.summary`, `palm`

Examples

```
library("PALM")
data("CRC_data", package = "PALM")
CRC_abd <- CRC_data$CRC_abd
CRC_meta <- CRC_data$CRC_meta

########## Generate summary statistics ##########
rel.abd <- list()
for(d in unique(CRC_meta$Study)){
  rel.abd[[d]] <- CRC_abd[CRC_meta$Sample_ID[CRC_meta$Study == d],]
}

null.obj <- palm.null.model(rel.abd = rel.abd)
```